SAFE – Secure Authentication in Federated Environment using CEG Key code
(A novel method to enhance cloud security)

Dr.B.Murugeshwari[1], K.Sudharson[2], S.P.Panimalar[3], M.Shanmugapriya[4], M.Abinaya[5]

[1] Professor & Head, [2,3,4,5] Assistant Professor
[1,2,3,4,5]Velammal Institute of Technology, Panchetti, ThiruvallurDist, Tamilnadu.

Article Info
Volume 82
Page Number: 4760 - 4765
Publication Issue:
January - February 2020

Article History
Article Received: *18 May 2019*
Revised: *14 July 2019*
Accepted: *22 December 2019*
Publication: *23 January 2020*

Abstract

Cloud computing is an efficient way to power the IT industry, however security is a major concern while storing data to the Cloud. Cloud data are always a threat for accuracy in cloud. While the cloud-related infrastructures are far more powerful and robust than personal computers, they still face the wide array of internal and external data integrity challenges. In this paper we propose, a secure cloud authentication system using Client End Generated (CEG) key code. Through this scheme, the data owner uploads the data in the cloud (implemented with Google Drive) and they are allowed to modify the data using the private key. To provide security to the cloud computing environment, RSA based Technique is used to generate key code using Mother board number of system, the Disk number of the system and the user password for authentication process based on a novel Encrypted User End Generated (EUEG) security algorithm which reduces the role of the third party and enhances the security check with automatic logging of malicious attempts and behaviour when the auditing is done by a Third Party Auditor(TPA)

Keywords; *EUEG , CEG, TPA, Cloud Security, Public Auditing*

I. INTRODUCTION

Cloud Computing uses the same technology, services and applications as the Internet, turning these kinds of into a self-service components. Cloud computing technology provides detailed deployment and implementation of the user system. Work applications on unspecified actual physical systems, the data is usually stashed in unknown places, the machine control is subcontracted in front of large audiences and user entry is ubiquitous.

Cloud processing is really a concept long imagined of calculation as the platform where users could store their data remotely in the cloud to allow them to enjoy the required applications and any services of good quality through a shared group of configurable resources.

However it makes data integrity protection in cloud computing a very difficult and potentially awesome task, particularly for users with limited resources and computer capabilities, because users no longer possess physical data, which could be large. Whenever the user/customer needs their data from the public cloud, they have to check the integrity/reliability of their data with the help of third party auditors, because security and reliability/integrity are the most important things in data storage of cloud area. Thus facilitates the public audit scheme for cloud data storage.

Cloud Computing technology is the one in which sharing of resources is achieved through virtualization. Infrastructure and treatment of the centralized infrastructure can be provided as needed, costs are calculated on a measured basis, the media can be activated and resources are agile to scalable.

Third-Party Auditor (TPA) is the protected one, who must fulfill the two necessary conditions:

By requiring local copies of data, TPA should be able to efficiently track cloud data storage and not impose a further Online Burden for cloud users.

No new privacy vulnerabilities should be created in the external auditing process. In this task, we use and combine the public key authentication homomorphism with random masking to reach the part of the audit of the information meets all the above requirements.

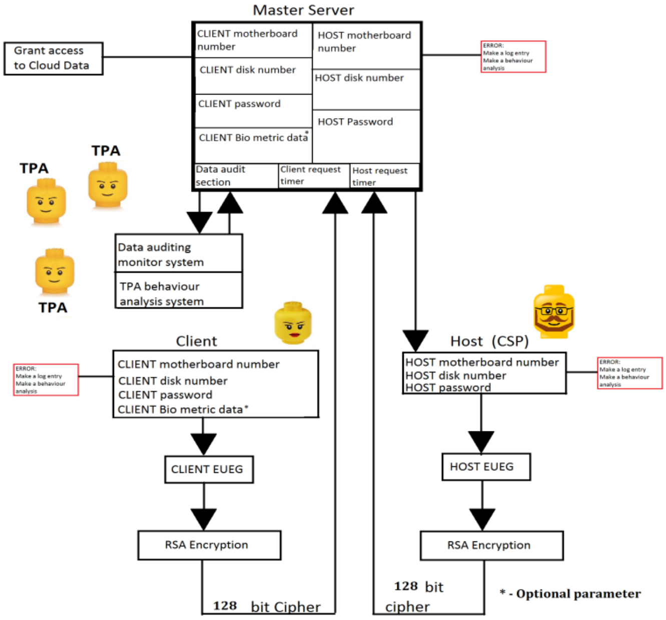
This technique ensures security of private/public cloud. Customers profit from premium quality providers in addition to save substantial purchases inside their own infrastructure by simply migrating local info administration systems on typically the impair servers.

II. RELATED WORK

The problem can be generalized in order to find an appropriate way to perform periodic integrity checking without copying data files locally, as in [1] [2]. If any two users or more users are using a data, one is writing a data while someone is reading a data then it may be wrongly read by a user, so as to resolve Data in-consistency, It is a major task of the data holder, who requires TPA but does not determine how to trust TPA. If the TPA becomes intruder and transmits data or deletes data, how the owner knows this is not resolved. The Advanced Encryption Standards is used in the encryption process and also decryption process of a file. With this virtual machine, this system solves the condition of unauthorized access of data. The proposed scheme works extremely well regarding the integrity and dependability of data. The TPA is proposed[5] to ensure data integrity via the source data and to protect the privacy, integrity and authenticity of outsourced information in cloud environments. It addresses the prior issue of allowing public verification and data storage processes of cloud computing as well as data protection auditing and collecting digital content in this way. TPA checks the quality of data on cloud on behalf of users. They usually cannot help with regeneration for two reasons like in[3]. First, because the incompatibility between the stored and calculated value of the checksums simply means that one of them has been altered but does not provide information as shown in[4], which is legitimate. Stored checksums may also be altered or damaged. Secondly, checksums are usually calculated using a single-way hash function, and data can not be recovered provided a checksum value.

We find that similar approach to detect the nodes based on the devices as in [6], but the problems caused by transmission jitter and network delays and their resulting anomalies are not well handled. Password settings for cloud resources (applications, virtual servers etc.) does not comply with user organization's password policies [7], it is also noted that the CSP in turn use several sub CSPs to support their Cloud Framework and technology. These Sub CSPs and the CSPs may have a security standard that may not be suitable for the end client. Terminated users found to be active on applications in the cloud (even though the individual's network access was terminated) and there was no IP range restriction , Employees transferred out of a certain department had access to Cloud resources even though they transferred to another department in the past work schedule[7]. Important legal notices and log notifications were submitted only to System administrators who in most cases are not included in the Company's board of control. It poses a high risk to the organization[7]. Juels and Kaliski[8] suggested a sentinel-based program called a POR for preserved files. A error-fixing code is applied and encrypted before an embedded number of sentinels is specified. The new file is then allowed to be sent to the server with sentinels. The drawback of this protocol is that the server can not delete any blocks and pretends that the entire file does not know that blocks are sentinels. The down side, of course is that the client can only challenge the server limited number of times. This critical constraint allows the protocol not appropriate for services which need dynamic checks and frequent changes. Further the major computation work happens in the client side, this is a huge burden on the resources of Client and leads to increased computation cost. In [9] Roskchke et al. discussed the architecture of IDS that can detect programs with malicious behavior. IDS monitor incoming and outgoing traffic and detect intrusions. Many management issues were aroused with respect to IDS. IDS have been enhanced to provide more security. The difference in them does not provide information about how malicious programs are treated by the IDS. Logging of malicious behaviours was not accounted in detail to analyze the anomalous pattern and prevent further occurrences. Moving on to the behaviour pattern analysis of Auditors and general algorithms found to detect behaviour, we find Cook et al. [11] initially focused on building universal models, represented by means of Markov models, to predict either future locations or activities. Improvements were made by developing applications to discover daily and weekly patterns [12]. This paper is extended by Jakkula and Cook[13], to predict action using tmporal-relationships defined by the temporal logic relations of Allen[14]. In order to assess if a given situation was normal or abnormal, Chan et al. developed a similar application[15]. The use of ANNs has a limitation related to their internal structure not being easily comprehensible; hence we proposed a system so as to remain a light weight cloud application on the client side. [16] gives an insight in forecasting security information and approximating frequent miniscule changes. This is used to project a pattern to identify the anomalous activity structure in the TPA behavior data. [17] illustrates event correlation analysis and detection of similar characteristic attributes. This aids in understanding common behavior from anomalies in TPA analysis.

III. PROPOSED SCHEME



Encryption of User End Generated Key

Registration phase:

We obtain the disk number and mother board number using a java client end module during the registration phase. We process the Alpha numeric motherboard number into a processed Mother board number. The processed Mother Board number consists of the ASCI representation of the respective characters in the Mother Board number.

The client is also requested to mandatorily enter three security questions and their corresponding answers in case of password recovery in future. These details are stored in the database.

We also get the user’s alpha numeric password. The CSP assigns a “ Secure Key “ for message encryption. For every three months this key is required to be changed. This Secure key is used to encrypt all authentication transactions regarding the EUEG authentication service scheme.

Algorithm (Working Principle):

We deploy a client end module. This module is capable of detecting the Mother Board number andprocess it to get the required ASCI format. It detects the disk number. It receives the user's password.

This Capsule follows the below algorithm :

Capsule = (DskNo + MoboNum) * UPsw

Notation:

DskNo = Disk Number detected on client end

MoboNum = Mother Board number on client end.

UPsw = the password entered by the user.

This capsule is prepared within a time frame of 30 micro seconds.

Dynamic Nature of EUEG:

The EUEG is time bound and dynamic to enhance the security aspect. The entire processing is completed within 120 seconds (2 minutes). The existing OTP system has a minimum waiting time of 5 minutes. We prepare a security parameter called "Time Capsule".

Time Capsule is generated by taking Capsule, Minute, Hour as the parameters.

Step 1: ProMin = (Minute +Prime) * (succeeding prime)^2 (for prime less than 10, to enhance the speed of processing in slow networks)

Step 2: ProHour = ProHour + (ProMin*2^n) (where 1<=n <=6)

Step 3: TimeCap = ProMin *ProHour*Capsule

Notations:

ProMin = Processed Minute

ProHour = Processed Hour

TimeCap = Time Capsule

We tend to use smaller limits of iteration for demonstrating in slow speed connections, however high speed ISDN lines can support higher iterations for processing the time capsule.

Encryption:

The TimeCap produced is now encrypted using RSA Algorithm. We use the chosen secret key to encrypt the TimeCap.

Merits:

RSA is one of the safest algorithms in today's world. However we provide an additional layer of discrepancy to the algorithm.

The TimeCap is dynamic in nature. Thus the parameter varies with time and never repeats. It is also encrypted using RSA. Therefore even if the RSA key is compromised, the TimeCap parameter is safe from espionage as TimeCap depends on Minute, Hour, Capsule (DiskNo, MotherBoard No, UserPassword)

Verification:

The encrypted TimeCap (initiator) is received on the (Automated controller) Third party side. The third party would pass the TimeCap to the service provider. The service provider would check the client's access permissions set by the administrator. Then the service provider would generate the "key" from its end. This key is sent to the automated controller (third party) to allow the client to access the cloud resources. Log entries are made for every attempt to log into the system.

Third Party Auditor Behaviour analysis:

Auditors are monitored continuously for every audit they pass. The data segments are classified according to their complexity. In general, the complexity is proportional to the period of audit. If the audit durations of an auditor is found to deviate from the general trends of the audit duration of other auditors over a period of time, then that particular auditor is black listed. The data segments are audited again to ensure correctness.

The positive aspect is that auditors cannot estimate the audit time required for a particular data segment in advance and the system takes into account, the dynamic audit pattern changes. For an instance, if the audit duration is displaying an overall shift, the auditors will not be blacklisted according to a fixed standard relating complexity and duration. But this overall shift can never be controlled by a small group of malicious auditors.

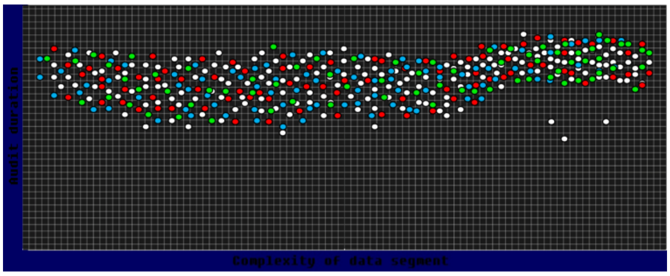
IV. EXPERIMENTAL IMPLEMENTATION

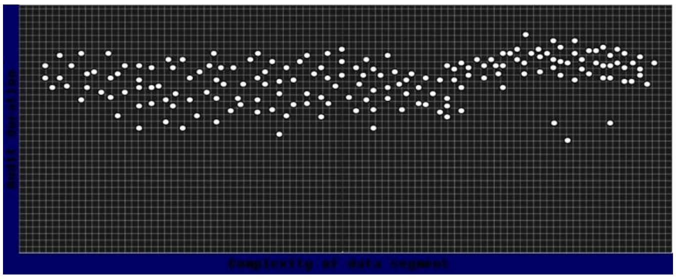
The EUEG was experimentally implemented in the institution's server, and 5 nodes running Windows 7 as their Operating System, 4 GB RAM and i3 processor were randomly selected as client.

This Client node on logging in, initiated the EUEG client module. An Encrypted User End Generated key (EUEG Key) was produced based on the motherboard number, disk number, user password, minute of log in and the hour of log in. The parameter was encrypted by means of RSA derived algorithm on the client module. This was later sent to the Master server for authentication. The Master server on receiving the EUEG(initiator) from the client end, intimated the CSP module to generate the EUEG (key) on its end. The EUEG received from the Client, the CSP were compared. This was verified within a time frame of 2 minutes. On verification the EUEG directed the client to their respective Google drive account as a proof of authentication.

We also had an attempt to enter the system by entering an invalid parameter, the authentication failed. Another attempt to enter the system by EUEG where the authentication procedure was extended for more than 2 minutes, the EUEG failed indicating the time window had expired. Various behavior of login attempts were done, all the results were stored in a database. These results were later used to form the training data to be used with Random Forest Algorithm [10] to detect anomalous behavior of client, CSP and TPA using their own training data set. Unusual behavior resulted in log entries that were stored in Master Server and had a read-only access limit to the Client, the CSP and TPA.

TPA behaviour patterns were implemented by testing the system with a set of four volunteers who acted as auditors. They were given files and asked to verify their checksums with a simulator. The simulator displayed longer check sums for complex files and shorter checksum for less complex files, their audit patterns were recorded with a custom simulator that was built containing JFreeChart to show the results in a graphical form. We represented the four auditors as Red, Green. White, Blue and based on the complexity of the file and the duration to validate the graph was plotted. It was found that the duration considerably increased for complex files. However towards the end, White auditor showed a deviation to general trends. Upon auditing some files selected by the white auditor again, we found that three of his files were audited in haste (improper auditing). This compromised the purpose of audit, and so he was black listed





V. PROPOSED FUTURE WORK

The proposed system has been implemented in the institution's servers. We have successfully registered the EUEG in the Google Developer's API access list, thereby gaining access to Google Drive (Cloud arm of Google Inc.) from our Java web application. However, we intend to implement the TPA behavior module in real time using Google

Drive. The preparations for the above real time implementation of TPA behavior modeling using Random Forest Algorithm are underway.

REFERENCES

[1] Ravi Kant Sahu, et al., “Robust Data Integration While Using Third Party Auditor For Cloud Data Storage Services” 2012, Vol 2, Issue 2, IJARCSSE.

[2] Qian Wang et al., “Enabling Public Auditability And Data Dynamics For Storage Security in Cloud Computing” in IEEE transactions on “Parallel And Distributed Systems” , vol. 22, no. 5, 2011.

[3] M.Ashah et al., ”Privacy-Preserving Audit And Extraction of Digital Contents”, 2011.

[4] FarzadSabahi,“Cloud Computing Security Threats and Responses” , IEEE conference. 2011.

[5]Govinda V et al., “Third Party Auditing For Security Data Storage in cloud through digital signature using RSA” 2012, IJASATR, vol4.

[6] Phil Lageschulte et al., Cloud Computing: Risks and Auditing, [7] A. Juels et al., “Pors: proofs of retrievability for large files,” in ACM conf, pg 584–597, NY, 2007.

[8] W. Enck, et al., "TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones", In Proc. OSDI, Canada, 2010.

[9] Cook et al., “How smart are our environments? An updated look at the state of the art,” in Pervasive and Mobile Computing, Amsterdam, The Netherlands: Elsevier Science, 2007, pp. 53–73.

[10] Heierman et al., “Improving home automation by discovering regularly occurring device usage patterns,” in Proc. Third IEEE conference, Nov. 2002, 537–540.

[11] V. R. Jakkula et al., “Using temporal relations in smart environment data for activity prediction,” in 2007. Proc. 24th Int. Conf. Machine Learning.

[12] J. Allen, “Toward a general theory of action and time,” Artif. Intell., vol. 23, pp. 123–154, Jul. 1984.

[13]M. Chan, C. Hariton, P. Ringeard, and E. Campo, “Smart house automation system for the elderly and the disabled,” in Proc. IEEE Int. Conf. Syst. Man Cybern., Oct. 1995, pp. 1586–1589.

[14] Song-song Lu, Xiao-feng Wang, Li Mao (2014) "Network security situation awareness based on network simulation"

[15] Zhang et al., "The Study of Network Security Event Correlation Analysis Based on Similar Degree of the Attributes", IEEE conf., China, 2013.

[16] D. Sánchez and M. Batet, “Privacy-preserving data outsourcing in the cloud via semantic data splitting,” Computer Communications, vol. 110, pp. 187–201, 2017.